Dublin Business School

excellence through learning

WEB & MOBILE TECHNOLOGIES ASSIGNMENT 01

CarHub

Module: Web and Mobile Technologies (B9IS124)
Module Leader: Mr. Ehtisham Yasin
Team Members: Manik Mahashabde (10518579)
Saurabh Devade (10531140)

Table of Contents

TABLE OF FIGURES

1. PROBLEM SCENARIO

You have been hired by a car adverts business, and have been tasked to develop a website that allows you to view a number of car adverts (develop a system to display at least 10 records each record should have multiple images, etc.). You will work on the layout of the website and allow for the user to navigate through various makes and models and find the car they're interested in. You should design the same for a mobile device as well. The user should be able to lookup the advert based on make, model and year of manufacture for the car, and view the details of each ad, using the print CSS. The user should be able to contact the dealer using a contact form if they're interested in the ad.

2. APPROACH TO THE PROBLEM

2.1. APPROACH: Our first step was to plan the website on a blank sheet. We decided how many pages should be there. How the layout of the page should be. What sections should be there? How the banner will be placed, which sections will be given more importance etc. Then we moved on to deciding the grid of the car details. How many cars should be displayed in each row, what should be the width of each grid etc.

The next step was to collect data for car adverts. We decided that we will have data of 20 Cars which will be stored in **JSON** *("JSON")* File. The brands which we selected are Mercedes Benz, BMW, Audi, and Bentley. We extracted the data for cars from official websites and found out some attractive images of each of these cars. We selected 5 cars of each of the 4 brands and 5 images of each car. *("Mercedes-Benz Passenger Cars", "Audi Ireland", "BMW Group: The Company, Careers and Investor Relations", "World of Bentley | News & Features | Bentley Motors")*.

After gathering the requirement and all the data, the next step was to choose the technologies and framework for developing a reliable, interactive and responsive website with ease. So we settled down for the following technologies and frameworks.

2.2. HTML5: We used HyperText Markup Language for structuring the layout of the webpages, inserting the content on the webpage, placing the images on the webpage, creating buttons, etc. *("W3C HTML")*

2.3. CSS3: To make the webpage look attractive we have used the latest version of Cascading Style Sheets which is CSS3. To make pages responsive i.e. it should adjust the content according to any viewport, we used CSS3's flexbox properties. Also, it has many features for various transitions for animation, etc. *("Cascading Style Sheets")*

2.4. JSON: JavaScript Object Notation is nothing but a data stored in a key-value pair. It is widely used because of its feature that it is human readable. We have also used ajax for fetching the data so JSON is the best way to retrieve data. *("JSON")*

2.5. AngularJs: We have used Angular Js which is a JavaScript framework for creating a dynamic web application. Using angular, we have used various directives like ng-repeat, ng-click, ng-src, etc. The advantage of angular is that it replaces a number of lines from JavaScript and JQuery by a single line. *("AngularJS — Superheroic JavaScript MVW Framework")*

2.6. AJAX: AJAX is a client-side asynchronous technology. It is used for communicating asynchronously with the server without changing the display and behavior of the existing webpage. *("Ajax")*

2.7. jQuery: We have also used javascript's framework jQuery. The purpose of using Jquery is to add the interactivity to the webpage like hide, show, performing some actions on click of particular div or button. *(JS Foundation - js.foundation)*

2.8. Materialize Framework: Materialize framework has many features like Grid management, Carousel, pop-ups, etc. But we have used materialize CSS specifically for adding a parallax effect to the webpage. *("About - Materialize")*

3. SITE MAP

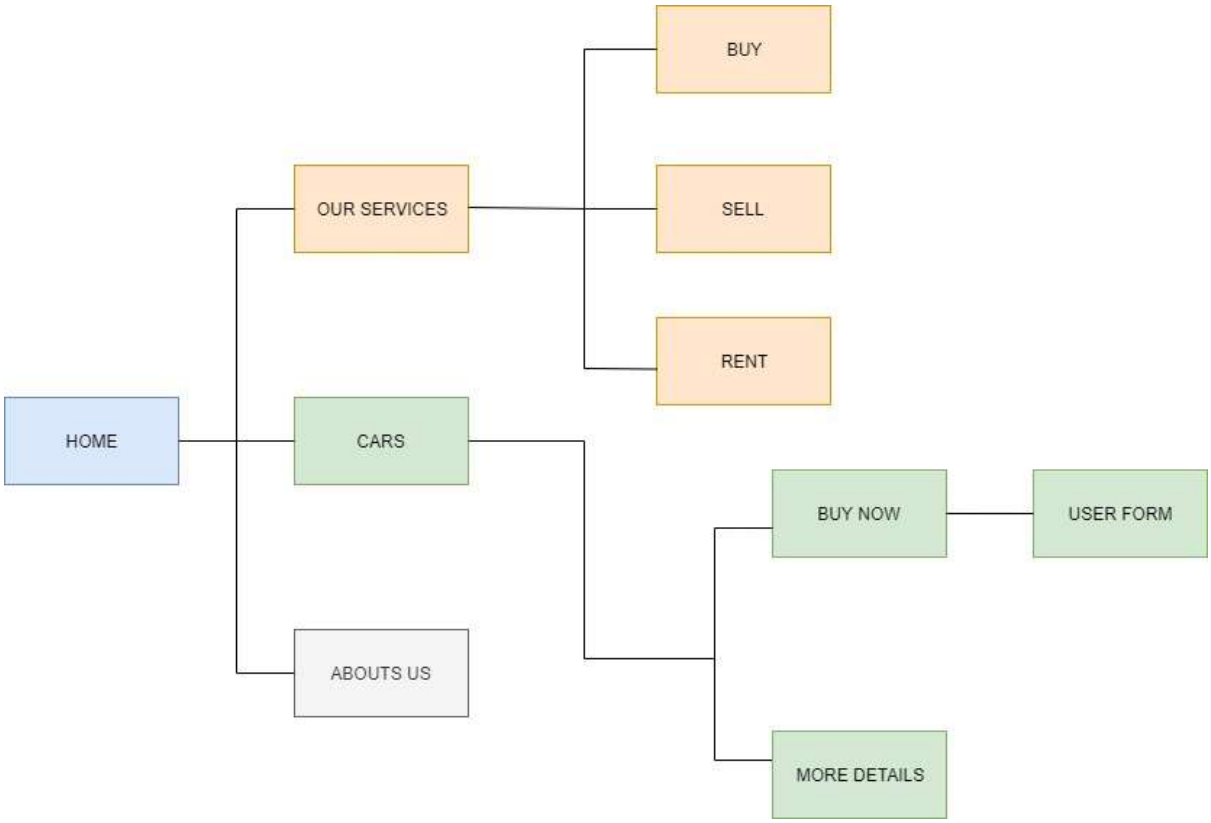

Figure 1 - Sitemap

4. APPROACH FOR DECIDING THE LAYOUTS FOR THE IMPLEMENTATION OF THE SCENARIO

4.1. HEADER: After deciding technologies and gathering all the required data and gathering the external frameworks we decided to build a **single** page application for our car adverts. Which contains the navbar at the top. Navbar is divided into two halves. The left section will contain the Logo and the right section has menus for navigating through the website. On the mobile view, dropdown is displayed on the clicking of the hamburger icon.

Figure 2 - Desktop Navbar

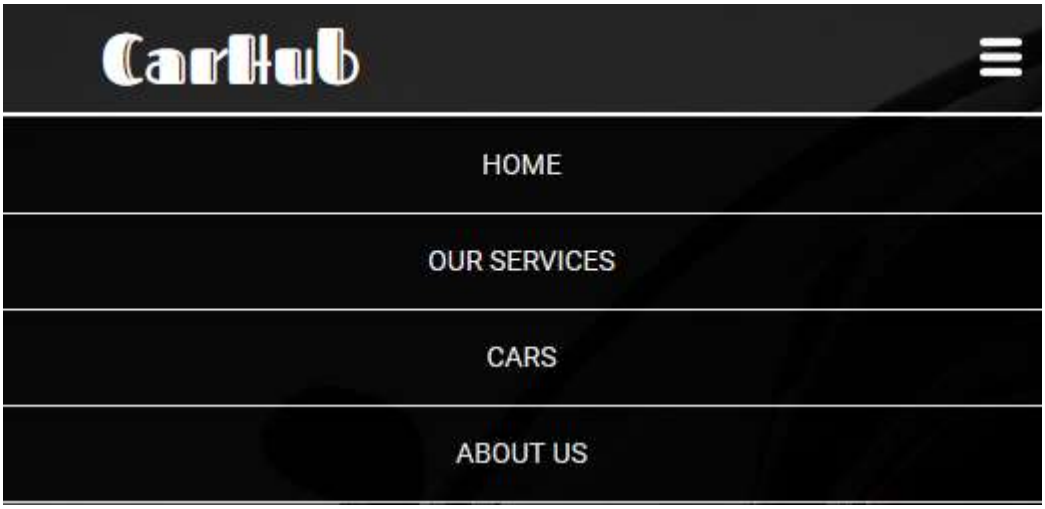

Figure 3 - Mobile Dropdown Menu

4.2. MAIN BANNER: As soon as visitors land on the webpage the first thing he notices is the banner. So we have used a full width and height banner. It will basically take all the viewport on any resolution device. To achieve this functionality we have used CSS's **background-image** property.

Figure 4 - Desktop Banner

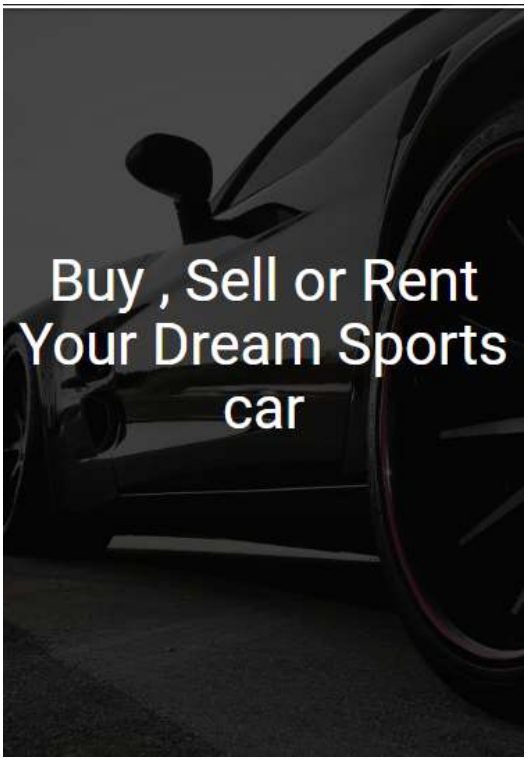

Figure 5 - Mobile Banner

4.3. OUR SERVICES SECTION: Then comes the our services section. Our Services sections display the services we offer which are BUY CARS, RENT CARS, SALE CARS, etc. We are displaying three div in a row. To make it responsive we have used flexbox property. On mobile resolution, all the divs will be one below other and will take a 100% width of a viewport.

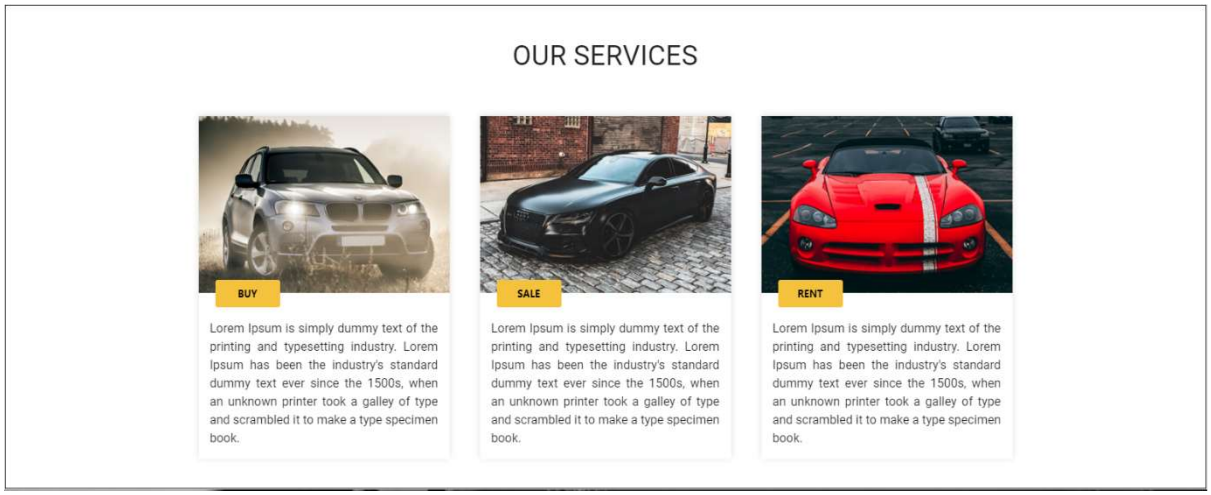

Figure 6 - Our Services

4.4. JSON DATA AND AJAX CALL: Before going further let's discuss how we have Stored JSON Data and fetched the required data from it.

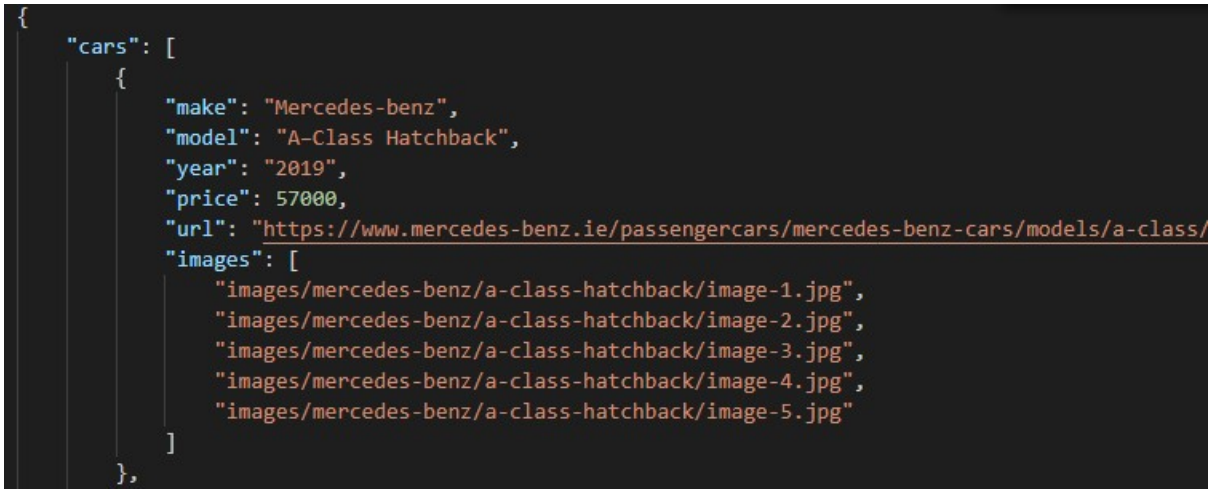

Figure 7 - Json Data

For retrieving data we are using angularjs ajax call using $http request.

```
var lvm = this;
lvm.getData = function () {

  $http({
    method: "POST",
    url: "json_data/adverts.json"


  }).then(function mySuccess(response) {

    lvm.cars = response.data.cars;
    lvm.filtredCars = lvm.cars;
    console.log(lvm.cars)

  }, function myError(response) {
    $scope.myWelcome = response.statusText;
  });
}

lvm.getData();
```

Figure 8 - AngularJs Ajax Call

We are using the post method and giving the relative URL of that file. once ajax call return success code all the retrieved data is stored in lvm.cars variable and further operation are performed on that.

4.4. OUR CARS SECTION & FILTERS SECTION: Then comes the main feature of the website which is displaying all the cars and filtering through the cars as per the need. To achieve this we have used different buttons like Make, Year and price range. On clicking of the button, a dropdown will be opened and clicking the specific option filters will be applied.

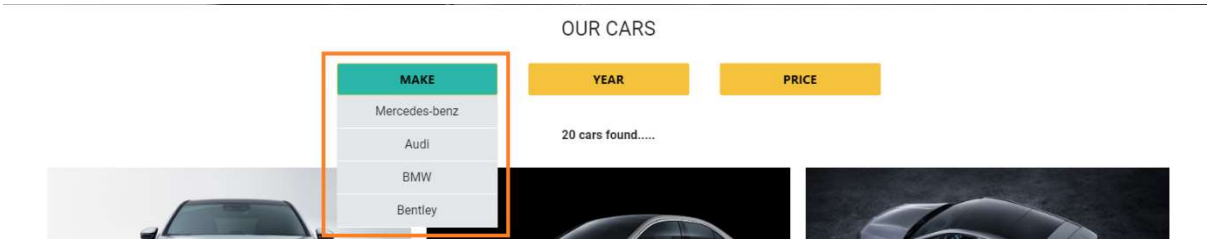

Figure 9 - Filter Buttons

On choosing the filter **lvm.applyfilterFunction(carMake)** function will be called, we have to pass carMake value to the function as well for eg. '**BMW'**. This function will match all the data stored in JSON against this carMake and will return all the selected cars whose make is **'BMW'.**

```
lvm.applyMakeFilter = function (carMake) {

  lvm.filtredCars = []
  if (carMake != null && carMake != undefined) {
    for (var i = 0; i < lvm.cars.length; i++) {
      if (lvm.cars[i].make === carMake) {

        lvm.filtredCars.push(lvm.cars[i]);
      }
    }
  } else any
    lvm.filtredCars = lvm.cars;
  }
  $("#carMake").toggle('slow');
}
```

Figure 10 - Filter Function

The data fetched from the JSON is displayed in 3 cars in a row using CSS flexbox. To repeat the HTML and CSS from the number of cars on the webpage we have used angularjs's ng-repeat directive.

```
<div class="cars-container">
    <div ng-repeat="car in lvm.filtredCars" class="grid-outer-contianer modal-trigger" data-target="modal1">
        <div ng-click="lvm.showSLiderData(car)">
            <img ng-src="{{car.images[0]}}" class="grid-iamges">

        </div>
        <div class="grid-item-data">
            <p>{{car.make}}</p>
            <p> {{car.model}}</p>
        </div>
    </div>
</div>
```

Figure 11 - ng-repeat directive

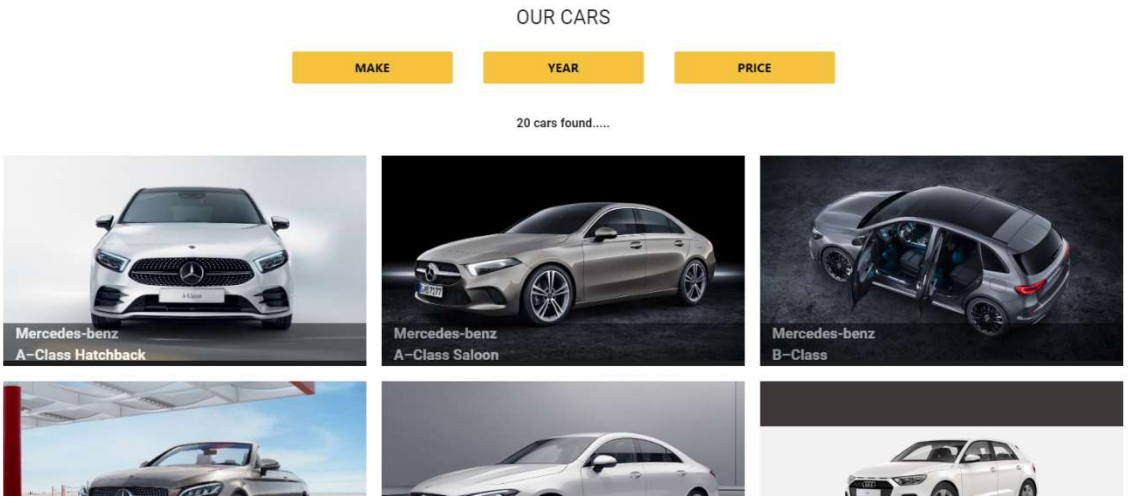

Figure 12 - Cars Grid

4.5. HOVER EFFECT AND POPUP: To add some effects, when a user hovers over a particular car CSS's **transform: scale(1.2)** property is used so it gives the zoom-in effect.

Figure 13 - Hower Effect

Once the visitor clicks on a car a popup will open and it will display a slider of all the car images.

Figure 14 - Popup

There are two buttons on the popup for BUY NOW and MORE DETAILS.

Once You click a Buy now button a form will open where the user can enter details and form is validated as well.

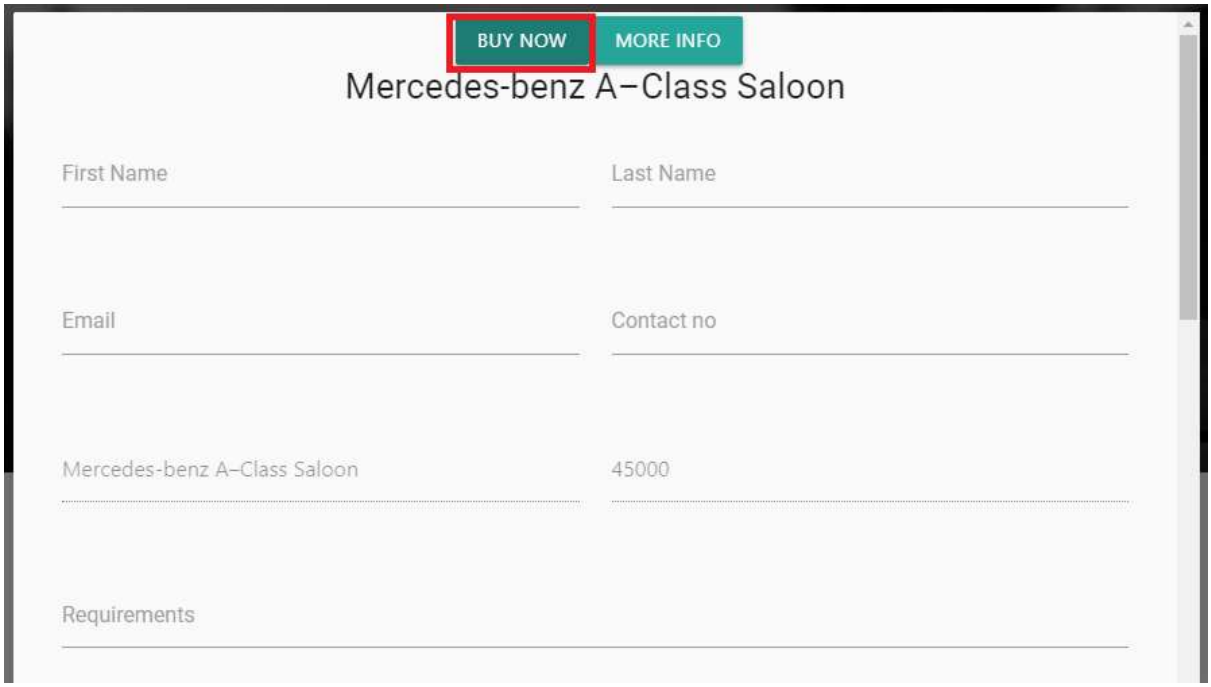

Figure 15 - Buy Now Form

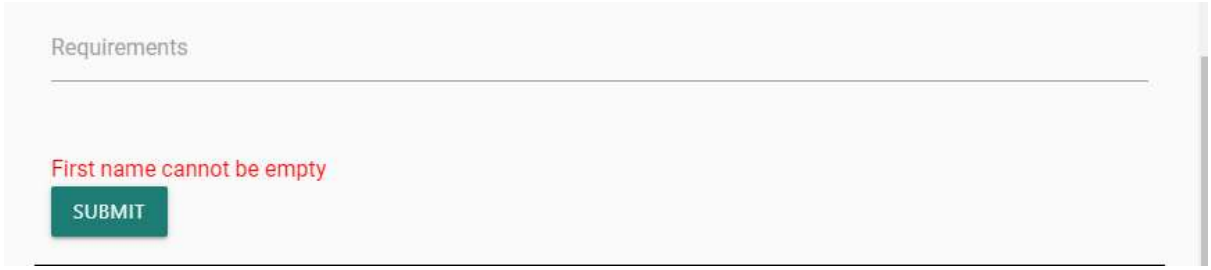

Figure 16 - Form Handling

4.6. ABOUT US & TESTIMONIAL SECTION: In about us section we have entered our company details and what it does whereas in the testimonial section the testimonial from previous clients includes which is very important to attract more clients.

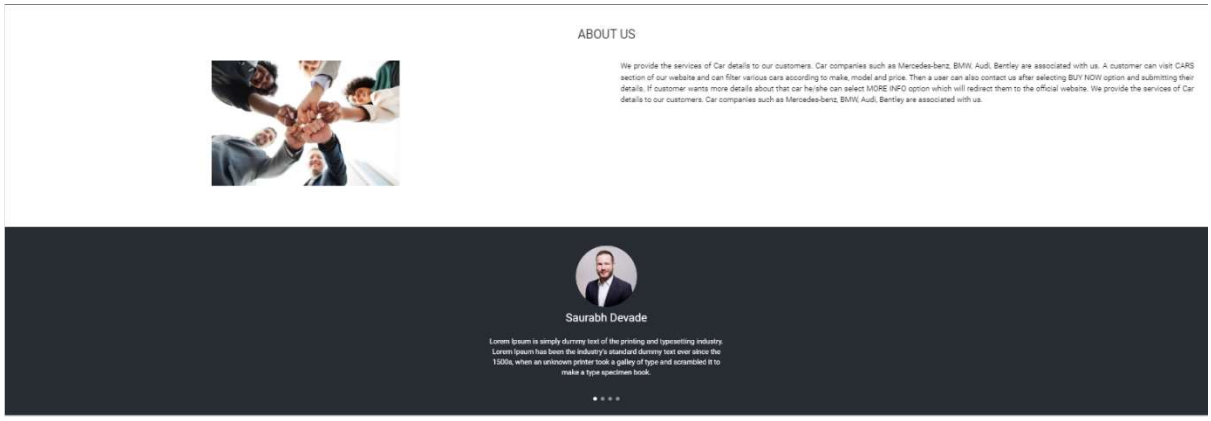

Figure 17 - About Us and Testimonials

In this manner by proper planning and decision making, we have completed our full-featured website.

5.REFERENCES & BIBLIOGRAPHY

"About - Materialize." *Materializecss.Com*, 2014, materializecss.com/about.html. Accessed 19 Nov. 2019.

"Ajax." *MDN Web Docs*, 25 July 2019, developer.mozilla.org/en-US/docs/Web/Guide/AJAX. Accessed 19 Nov. 2019.

"AngularJS — Superheroic JavaScript MVW Framework." *Angularjs.Org*, 2010, angularjs.org/. Accessed 23 Apr. 2019.

"Audi Ireland." *Audi.Ie*, 2019, www.audi.ie/ie/web/en.html. Accessed 19 Nov. 2019.

"BMW Group: The Company, Careers and Investor Relations." *Www.Bmw.Ie*, 2019, https://www.bmw.ie/en/topics/fascination-bmw/bmw-group.html. Accessed 19 Nov. 2019.

"Cascading Style Sheets." *W3.Org*, 2019, https://www.w3.org/Style/CSS/Overview.en.html. Accessed 2 Nov. 2019.

JS Foundation - js.foundation. "JQuery." *Jquery.Com*, 2019, jquery.com/.
"JSON." *Json.Org*, 2019, www.json.org/. Accessed 19 Nov. 2019.

"Mercedes-Benz Passenger Cars." *Mercedes-Benz.Ie*, 2019, https://www.mercedes-benz.ie/passengercars.html. Accessed 19 Nov. 2019.

"W3C HTML." *W3.Org*, 2018, www.w3.org/html/.

"World of Bentley | News & Features | Bentley Motors." *Bentleymotors.Com*, 2019, https://www.bentleymotors.com/en/world-of-bentley.html. Accessed 19 Nov. 2019.